Exercises on 2D Arrays (Matrices)

Create a lab9 folder to hold the files for this assignment. When you are finished, compress the directory and submit it on Moodle.

To start, create a new Java file called MatrixTester.java. This file will hold methods to perform operations on 2-dimensional arrays of ints. Your MatrixTester class should contain a main method that runs tests of all the methods in the class.

Exercise 1: Write a method called addMatrixes that consumes two n x m matrixes A and B of type int[ ][ ] as parameters. Return the new matrix A+B of size n x m. You can assume that the matrixes are of the same dimension (see if you can figure out how to check if the matrices are of the same dimensions...if not, throw a RunTimeException). Your output should look like that shown below (but you should test it on different size matrices):

Matrix A:
 3 4 5
 2 3 6
 5 3 1

Matrix B:
 1 2 3
 4 5 6
 1 1 1

Matrix A+B:
 4   6   8
 6   8 12
 6   4   2

Exercise 2: Suppose that M is a two-dimensional array that has R rows and C columns. The transpose of M is defined to be an array T that has C rows and R

columns such that `T[i][j]` = `M[j][i]` for each `i` and `j`. Write a method called transposeMatrix that takes an array of type `int[][]` as a parameter, and returns the transpose of that matrix. Assume that the parameter is a typical 2D array in which all the rows have the same length (see if you can figure out how to check if the row length is the same for all rows of M...if not, throw a new RuntimeException). Your output should look like that shown below:

Matrix R:
 1 2 3 4
 4 3 2 1

Transpose of Matrix R:
 1 4
 2 3
 3 2
 4 1

Exercise 3: Write a helper method called printMatrix that takes as input a 2-dimensional array and prints it out like those shown above. Use the System.out.printf("%2d ", M[i][j]) to make sure the columns line up neatly. The number after the % will vary, depending on the size of the numbers in the matrix.